A Regular Fast Multipole Method for Geometric Numerical Integrations of Hamiltonian Systems

P. Chartier · E. Darrigrand · E. Faou

Abstract The Fast Multipole Method (FMM) has been widely developed and studied for the evaluation of Coulomb energy and Coulomb forces. A major problem occurs when the FMM is applied to approximate the Coulomb energy and Coulomb energy gradient within geometric numerical integrations of Hamiltonian systems considered for solving astronomy or molecular-dynamics problems: The FMM approximation involves an approximated potential which is not regular, implying a loss of the preservation of the Hamiltonian of the system. In this paper, we present a regularization of the Fast Multipole Method in order to recover the invariance of energy. Numerical tests are given on a toy problem to confirm the gain of such a regularization of the fast method.

1 Introduction

In applications to molecular dynamics, the evaluation of the potential function involves pairwise interactions (the Coulomb energy) of a large number of particles (typically millions of) and constitutes the bulk of computations. In this context, different strategies have been considered to speed up its evaluation: One of them consists in using the Fast Multipole Method (FMM) as introduced by L. Greengard and V. Rokhlin ([12]). This first version was written to deal with point charges, while in papers [11], [10], [9], [7] and [5], the FMM was extended to and developed for the case of continuous distributions of the charges which corresponds to the charge distributions in molecular dynamics. These improvements led to versions of the FMM referred to as the continuous or gaussian Fast Multipole Method

P. Chartier · E. Darrigrand · E. Faou
Equipe projet IPSO, INRIA Rennes and Ecole Normale Supérieure de Cachan - Antenne de Bretagne, Avenue Robert Schumann, 35170 Bruz, France
E-mail: chartier@irisa.fr , eric.darrigrand-lacarrieu@univ-rennes1.fr , efaou@irisa.fr

E. Darrigrand
IRMAR - Université de Rennes 1, Campus de Beaulieu, 35042 Rennes cedex, France

(CFMM or GFMM). However, all these versions share the drawback of producing discontinuities which in turn, when used in combination with a symplectic time discretization of the Hamiltonian dynamics, lead to drift in energy. In molecular dynamics however, it is crucial that the numerical method used to compute the solution preserves the symplecticity, the volume form, the Hamiltonian, or a combination of the three (given that for smooth Hamiltonians, symplecticity implies preservation of volume) and for these properties to show up in long-term integration, quite a lot of smoothness is required. Ben Leimkuhler's work on smooth switches between different symplectic integrators points toward the same direction [4,3]. In order to attenuate the effect of discontinuities, one may use a large (sometimes up to twenty) number of multipoles in FMM expansions at the price of a prohibitive increase of the computational cost, despite many efforts to reduce the complexity of the FMM (see Elliot and Board [14], Petersen *et al.* [15], Scuseria *et al.*[5–7], and [8,13]).

In this paper, we propose a regularization technique for the FMM with the aim of recovering the usual benefits of symplectic integration: This new FMM is regular at the interface between boxes and thus provides a **smooth** approximation of the potential. The following section introduces the Fast Multipole Method and emphasizes its major drawbacks. The Regular FMM (referred to as the RFMM in the sequel) is then presented in Subsection 2.2. Finally, an application of the RFMM and numerical comparisons with the classical FMM are given in Section 3 where both methods are applied in combination with the Velocity Verlet scheme.

2 Regularization of a Fast Multipole Method

2.1 The Fast Multipole Method

We now briefly describe the Fast Multipole Method as derived in [12,13]. We consider a one-level algorithm and describe the regularization technique in a simple case. The extension to improved version of the FMM, see [14,15,8,13], can be performed similarly.

2.1.1 The FMM expansion for a standard kernel

Let A be the matrix defined by:

$$\forall i, j \in \{1, ..., N\}, \quad A_{ij} = \frac{1}{\|x_i - x_j\|}$$

where $x_1, ..., x_N$ are N points of a bounded domain D of $\mathbb{R}^3$ and $\|\cdot\|$ denotes the Euclidean norm on $\mathbb{R}^3$. For a given vector $q = (q_j)_{j=1}^N$, the computation $(Aq)_i = \sum_{j=1}^N \frac{1}{\|x_i - x_j\|} q_j$ for a given i, involves the *target* point x_i and the *source* points $\{x_j\}_j$. Let $\mathscr{B}$ be a partition of the domain D into groups containing the points x_j, $j = 1, \ldots, N$. We denote by $B_{\cdot}$ (for example: B_{src}, B_{trg}) the elements of $\mathscr{B}$ and $C_{\cdot}$ (for example: C_{src}, C_{trg}) their centers.

To approximate the computation of $(Aq)_i$, we split the summation as follows:

$$(Aq)_i = \sum_{B_{src} \text{ close to } B_{trg}} \sum_{x_j \in B_{src}} \frac{1}{\|x_i - x_j\|} q_j \ + \sum_{B_{src} \text{ far from } B_{trg}} \sum_{x_j \in B_{src}} \frac{1}{\|x_i - x_j\|} q_j,$$

where B_{trg} denotes the group which contains the target point x_i. In the FFM approximation, the first term is computed exactly, while the second term is approximated thanks to the

multipole expansion and a translation operator which converts multipole expansions to local expansions (for details, see Appendix 4.1 or [12], [13]).

The algorithm for the calculation of the matrix-vector product Aq can be summarized as follows (see Appendix 4.1 for complete details):

- Step 0: Calculation of some q-independent quantities:
  - · Parts of the translation operator $\mathcal{T}_{B_{trg}B_{src}}$ that converts a multipole expansion around C_{src} to a local expansion around C_{trg}.
  - · The far moments $f_j^{l,k}$ and the local moments $g_i^{l,k}$ involved in the multipole and local expansions, for each source points x_j or target points x_i.
- Step 1: Calculation of the far fields: For all B_{src} in $\mathcal{B}$, $F_{B_{src}}^{l,k}$ accumulates the charges q_j together with the far moments $f_j^{l,k}$ of the source points contained in B_{src}.
- Step 2: Calculation of the local fields: For all B_{trg} in $\mathcal{B}$, $(G_{B_{trg}}^{l,k})_{l,k}$ is obtained by translations of the far fields $(F_{B_{src}}^{\lambda,\kappa})_{\lambda,\kappa}$ for all B_{src} in $\mathcal{B}$ far from B_{trg}.
- Step 3: Accumulation of the far interactions: For all B_{trg} in $\mathcal{B}$, for all x_i in B_{trg}, $(Aq)_i^{far}$ accumulates the local fields $(G_{B_{trg}}^{l,k})_{l,k}$ together with the local moments $(g_i^{l,k})_{l,k}$.
- Step 4: Calculation of the close interactions: For all B_{trg} in $\mathcal{B}$, for all x_i in B_{trg}, $(Aq)_i^{close}$ accumulates the contribution of the neighbor source points x_j.
- Step 5: Calculation of matrix-vector product: For all B_{trg} in $\mathcal{B}$, for all x_i in B_{trg},

$$(Aq)_i \approx (Aq)_i^{close} + (Aq)_i^{far}. \tag{1}$$

In this algorithm, the interactions $x_i \leftrightarrow x_j$ are cut into a succession of interactions involving the groups containing x_i and x_j. Such an approximation obviously introduces discontinuities in the approximated quantity $\frac{1}{\|x_i - x_j\|}$ for both variables x_i and x_j, whenever x_i or x_j crosses the interface between two groups. Figure 1-a (respectively Figure 1-b) shows the critical situation where two target points (respectively two source points) are very close to each other but are not in a same group.

2.1.2 The FMM expansion for the gradient of a standard kernel

When applied to the numerical simulation of Hamiltonian systems involving a multi-body interaction potential, standard numerical schemes require the evaluation of the derivative of functions of the form $G(x,y) = \frac{1}{\|x-y\|}$. The FMM approximation of the kernel $\nabla_x G(x,y)$ is derived from the FMM approximation of the kernel $G(x,y)$. Indeed, the only difference lies in the calculation of the local moment which is not $g_i^{l,k}$ any longer, but the gradient of this quantity defined in the algorithm above. The FMM approximation of the kernel $\nabla_x G(x,y)$ is then based on the derivation of the quantity $g_i^{l,k}$ with respect to x_i which is made straightforward by noticing that the quantity $g_i^{l,k}$ is polynomial w.r.t $(x_i - C_{trg})$. Details are given in Appendix 4.2.

2.2 A Regularized Version of the Fast Multipole Method

As explained in Subsection 2.1, we apply a regularization technique to a rather basic one-level Fast Multipole Method. However, it can be easily extended to improved versions as considered in [14], [15], [8], [13]. At the end of this subsection, we discuss the multilevel
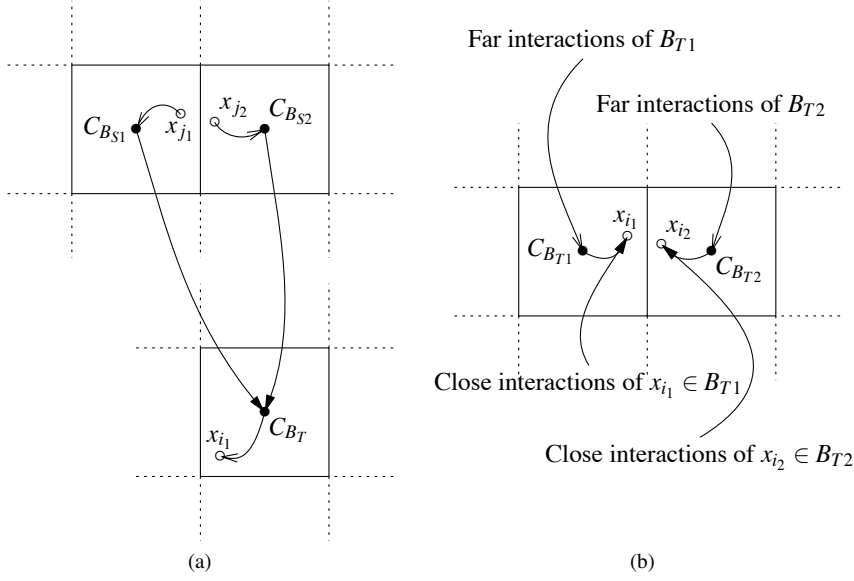
Fig. 1 (*generated with Fig4TeX*) Exhibition of the discontinuity in the approximation of the interaction between some couples of target and source points. (a) Two source points x_{j_1} and x_{j_2}, close to each other, interact differently with the target point x_i. (b) Two target points, close to each other, receive differently the information from their environment.

version. Besides, the multi-dimensional regularization is obtained by considering a 1D regularization on each component of the multi-dimensional variable.

In this subsection, we mainly focus on the 1D regularization of the FMM approximation of the kernel $G(x,y)$. The regularized FMM approximation of its derivatives is obtained considering the derivatives of the regularized FMM approximation of the kernel.

2.2.1 A 1D regularization

As pointed out in Subsection 2.1, the FMM introduces a discontinuity related to the distribution of the domain points into groups. Such discontinuities appear at the interface between two groups. In this paper, we perform the regularization with the simple following idea: when a point of a group is close to another group, we view it as a shared point and its contribution to each group is calculated according to its location. This leads to a new distribution of the points. We call groups associated to this distribution "virtual groups".

In 1D, the groups in $\mathscr{B}$ are disjoint intervals. Let $[p_i, p_{i+1}]$, $i \in \mathbb{Z}$ define the i-th *geometric* group belonging to $\mathscr{B}$. For all i, let p_i^l and p_i^r be two points around p_i. We define the i-th *virtual* group as the interval $[p_i^l, p_{i+1}^r]$. Now for a given point $x \in \mathbb{R}$, we associate virtual groups and weights depending on the position of x with respect to these virtual groups. For example in Figure 2, the point x_1 belongs to group 2 and its weight is 1 and in this case, the relation (1) remains unchanged. Point x_2 belongs to groups 2 and 3, with weights respectively equal to c_2 and c_3 such that:

* $c_2 + c_3 = 1$; obviously, $c_2 > c_3$.
* c_2 and c_3 are given by a regularizing function $\chi : [p_3^l, p_3^r] \to [0,1]$,
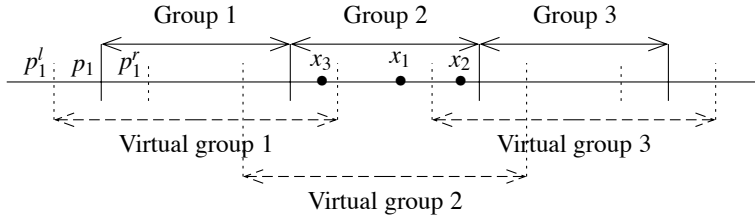
Fig. 2 (*generated with Fig4TeX*) The virtual groups for a regularized FMM: when the geometric groups are next to each other, the virtual groups overlap like in the partition of unity technique.

$c_2 = (1 - \chi(x_2))$, $c_3 = \chi(x_2)$ such that (1) becomes

$$(Aq)_{x_2} \approx (1 - \chi(x_2)) \; [(Aq)^{close}_{x_2 \in \text{ group } 2} + (Aq)^{far}_{x_2 \in \text{ group } 2}]$$
$$+ \quad \chi(x_2) \; [(Aq)^{close}_{x_2 \in \text{ group } 3} + (Aq)^{far}_{x_2 \in \text{ group } 3}]$$

Point x_3 belongs to groups 1 and 2, with weights respectively equal to c_1 and c_2 such that:

* $\star$ $c_1 + c_2 = 1$; obviously, $c_1 < c_2$.
* $\star$ c_1 and c_2 are given by a regularizing function $\chi : [p_2^l, p_2^r] \to [0, 1]$,
  $c_1 = (1 - \chi(x_3))$, $c_2 = \chi(x_3)$ such that (1) becomes

$$(Aq)_{x_3} \approx (1 - \chi(x_3)) \; [(Aq)^{close}_{x_3 \in \text{ group } 1} + (Aq)^{far}_{x_3 \in \text{ group } 1}]$$
$$+ \quad \chi(x_3) \; [(Aq)^{close}_{x_3 \in \text{ group } 2} + (Aq)^{far}_{x_3 \in \text{ group } 2}]$$

As discussed in the introduction, the aim is to get a C^∞ FMM approximation. The regularizing function χ should be chosen such that χ equals to 0 at the left extremity of the segment $[p_i^l, p_i^r]$, equals to 1 at the other extremity and such that all its derivatives vanish at both left and right extremities. Without any loss of generality, we can focus on the segment $[0, 1]$. A simple linear change of variable gives the function on any segment $[p_i^l, p_i^r]$.

Of course, many choices exist for the regularizing function χ. In the following, we use the function defined on $[0, 1]$ by the formula

$$\chi(x) = \frac{1}{1 + e^{-\alpha[1/(1-x^2) - 1/(1-(x-1)^2)]}} \tag{2}$$

The coefficient α influences the maximum gradient, which has to be taken minimal to avoid possible instabilities in the numerical schemes (presence of high eigenvalues).

Figure 3 gives the maximum gradient of the function versus the value of α. In this way, we obtain a good guess of α minimizing the maximum gradient of χ. In Subsection 3.2, we take $\alpha = 1.37$. Figure 4 shows the regularizing function for some values of α while Figure 5 shows the gradient and the second derivative for the same values of α.

2.2.2 Algorithmic consequences

The major consequence of the regularization is an enlargement of the FMM groups. It is clear that the regularization is improved when virtual groups are large in comparison with geometric groups (if p_i^l and p_i^r are too close to p_i, the regularization will be very close to a jump). However, one should keep in mind that the overlapping of groups may lead to
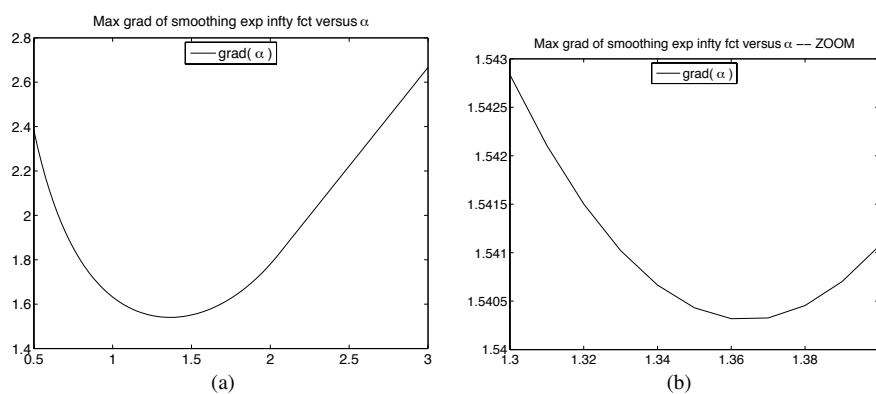
Fig. 3 (a) The maximum gradient of the regularizing function versus α. (b) Zoom around the optimal value of α.
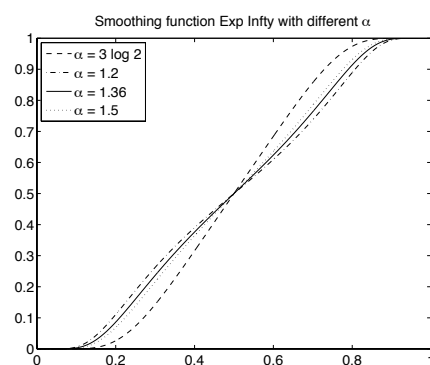
Fig. 4 Plot of the regularizing functions for some values of α ($3\log 2$, 1.2, 1.36, 1.5).
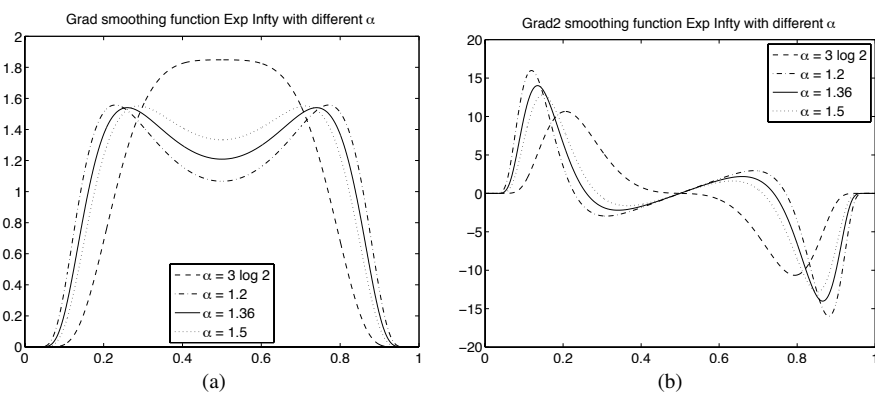
Fig. 5 (a) Plot of the first derivative of regularizing functions for $\alpha = 3\log 2$, 1.2, 1.36, 1.5. (b) Plot of the second derivative of regularizing functions for $\alpha = 3\log 2$, 1.2, 1.36, 1.5.

divergence of the multipole and local expansions (7) and (9). In order to avoid this to happen, we may have to consider a wider definition of the neighboring groups according to the size of the overlap. Indeed, in classical applications of FMM, two boxes B_1 and B_2 are said to be "close to each other" whenever they have at least one common vertex: We say that they are neighbors of order 1 and write $B_1 \overset{1}{\sim} B_2$. Here, we also may have to consider B_1 and B_2 neighbors of order 2: We write $B_1 \overset{2}{\sim} B_2$ if there is at least one box B_3 such that $B_1 \overset{1}{\sim} B_3$ and $B_3 \overset{1}{\sim} B_2$. The order of a neighborhood can be numerically characterized using the infinity norm when the groups are cubes from a usual FMM oc-tree.

The regularization can be performed for both the first and the second variables of the kernel $\frac{1}{\|x_i - x_j\|}$. However, in usual situations, one is interested in the regularization for the target variable only. When the regularization occurs on the source point, the contribution coefficient applies in the calculation of the far fields (17) in the FMM algorithm and only the source points are distributed in overlapping groups. When the regularization occurs at target point, the contribution coefficient is involved in the last step (1) and only the target points are distributed in overlapping groups. Here, we are interested in a regularization for target points only.

An increase of the FMM computational cost is expected due solely to the fact that some target points belong to two groups: it stems from the last step of the computation of far interactions (Step 3, relation (19)) and from the computation of close interactions (Step 4, relation (20)). The costs of steps 3 and 4 is indeed multiplied by the ratio between the average number of points in the virtual groups and the average number of points in the geometric groups; nevertheless, the complexity of these steps remains the same and the complexity of the whole algorithm unchanged.

2.2.3 Error estimates and 1D illustration

The FMM approximation induces an error in the evaluation of the potential function (see Eqn. (8) and (12) below). For the RFMM method, such estimates are still valid. Moreover, thanks to the regularity of the approximated potential, error estimates might still hold true in stronger norms. In this introductory paper, we deliberately leave these questions aside and rather illustrate *numerically* the improvement gained from the RFMM: As an example, we consider here a set of 800 points $x_1, ..., x_{800}$, uniformly distributed on the 1D domain $[0,1]$. For this illustration, we compute a vector S defined by: for $i = 1, ..., 800$, $S_i = \sum_{j=250}^{400} G(x_i, x_j)$ with $G(x_i, x_j) = \frac{1}{\|x_i - x_j\|}$ if $i \neq j$ and 0 otherwise. The quantity S can be defined as a classical matrix-vector product communly computed with the FMM. Here, we computed S in three different ways: exact calculation, using a classical FMM and using a regularized FMM. Figure 6 gives plots of the error obtained when using either the classical FMM or the regularized FMM. The figure clearly shows the effect of the regularization at the interfaces between the groups, where the classical FMM has evident discontinuities.

2.2.4 Extention to multilevel algorithms

In the one-level FMM, the regularization is only involved in the accumulation of far interactions from local fields. In the multilevel method though, steps 2 and 3, which compute local fields from far fields and accumulate far interactions from local fields are performed as follows:
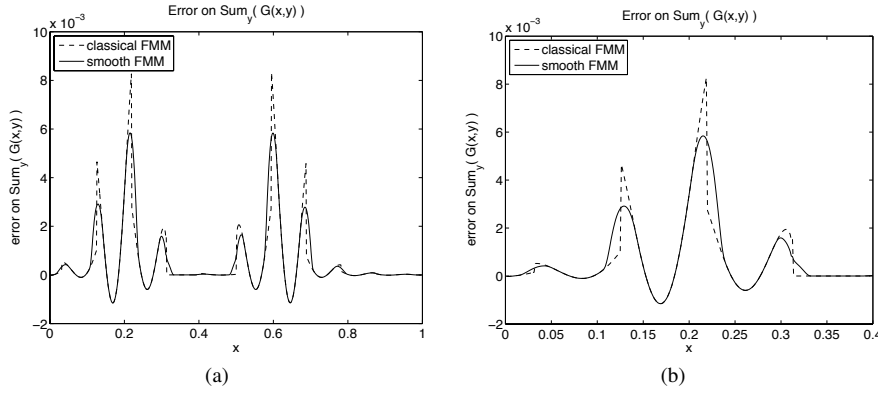
Fig. 6 (a) Plot of the error obtained using either the classical FMM (dashed line) or the regularized FMM (continuous line). (b) Zoom.

- Step 2: Calculation of the local fields at the coarser level: translation between all couples of groups.
- Step 2-bis: Calculation of the local fields at all the levels: translation between all couples of groups which are not neighbor but satisfying the condition that their parents are close to each other + translation of the information accumulated at the previous level (to a group from its parent).
- Step 3: Accumulation of the far interactions at the finest level only, taking into account the fact that all local fields at all levels have been accumulated at the finest level through Steps 2 and 2-bis.

Note that close interactions are taken into account only at the finest level. It is then clear that the same regularization can be applied to the multilevel algorithm involving the regularization in Step 3 and the calculation of the close interactions. The influence on the algorithmic complexity is similar to the one-level FMM situation.

3 Velocity Verlet Scheme and Regular Fast Multipole Method

In this Section, we apply the Verlet's method to a somehow "academic" model of the Outer Solar System as defined in [2]. Generally speaking and for a separable Hamiltonian system of the form

$$\begin{cases} \dot{q} = M^{-1}p \in \mathbb{R}^{3N} \\ \dot{p} = -\nabla U(q) \in \mathbb{R}^{3N} \end{cases} \tag{3}$$

where $M = \mathrm{diag}(m_1 I_{\mathbb{R}^3}, \cdots, m_N I_{\mathbb{R}^3})$ and with Hamiltonian $H(p,q) = T(p) + U(q)$, $T(p) = \frac{1}{2}p^T M^{-1}p$ being the kinetic energy and $U(q)$ the potential function, the Verlet's method reads (see for instance [2])

$$\begin{cases} q_{n+\frac{1}{2}} = q_n + \frac{h}{2}v_n \\ v_{n+1} = v_n - h\nabla U(q_{n+\frac{1}{2}}) \\ q_{n+1} = q_{n+\frac{1}{2}} + \frac{h}{2}v_{n+1} \end{cases} \tag{4}$$

where q_n and v_n denote approximations of $q(nh)$ and $v(nh)$ with $v = \dot{q} = M^{-1}p$. It is explicit, symplectic and symmetric, and preserves a modified energy for exponentially long time when the potential U is smooth.

Table 1 Data for the Outer Solar System

planet/Sun	mass	initial position	initial velocity
Sun	$m_0 = 1.00000597682$	(0 , 0 , 0)	(0 , 0 , 0)
Jupiter	$m_1 = 0.000954786104043$	-3.5023653 -3.8169847 -1.5507963	0.00565429 -0.00412490 -0.00190589
Saturn	$m_2 = 0.000285583733151$	9.0755314 -3.0458353 -1.6483708	0.00168318 0.00483525 0.00192462
Uranus	$m_3 = 0.0000437273164546$	8.3101420 -16.2901086 -7.2521278	0.00354178 0.00137102 0.00055029
Neptune	$m_4 = 0.0000517759138449$	11.4707666 -25.7294829 -10.8169456	0.00288930 0.00114527 0.00039677
Pluto	$m_5 = \frac{1}{1.3 \cdot 10^8}$	-15.5387357 -25.2225594 -3.1902382	0.00276725 -0.00170702 -0.00136504

When the scheme is applied in astronomy or molecular dynamics, the bulk of computations lies in the evaluation of ∇U for successive steps (order N^2). It is thus natural to consider its FMM approximation.

3.1 The RFMM for the Outer Solar System

The Outer Solar System (as defined in [2]) is the Solar system where the Sun is aggregated with the four closest planets (Mercure, Venus, Earth, Mars) and is modeled by Hamiltonian equations with

$$T(p) = \frac{1}{2} \sum_{i=0}^{5} \frac{1}{m_i} p_i^T p_i \text{ and } U(q) = -\gamma \sum_{i=1}^{5} \sum_{j=0}^{i-1} \frac{m_i m_j}{\left\| q_i - q_j \right\|}, \tag{5}$$

where γ is the gravitational constant. Denoting $G(x,y) = \frac{1}{\|x-y\|}$, the k-th component of the $(\mathbb{R}^3)^N$-vector $\nabla U(q)$ is given by the formula

$$\nabla_k U(q) = -\gamma \sum_{j \neq k} m_k m_j \nabla_x G(q_k, q_j) = -\gamma \sum_j \mathcal{M}_{k,j} \tag{6}$$

with $\mathcal{M}_{k,j} = m_k m_j \nabla_x G(q_k, q_j)$ for $k \neq j$ and $\mathcal{M}_{j,j} = 0$. This corresponds to a matrix-vector product commonly computed with the FMM.

3.2 Numerical results

The initial values are taken from [2] and listed in Table 1 (m_0 includes the mass of inner planets). Distances are expressed in astronomical units = UA (1 UA = 149,597,870 km), times in days, and $\gamma = 2.95912208286 \cdot 10^{-4}$.

Function (2) is implemented with the value $\alpha = 1.37$ suggested by Figure 3. The time-step is chosen equal to 10 days. The size of the boxes should take into account the length
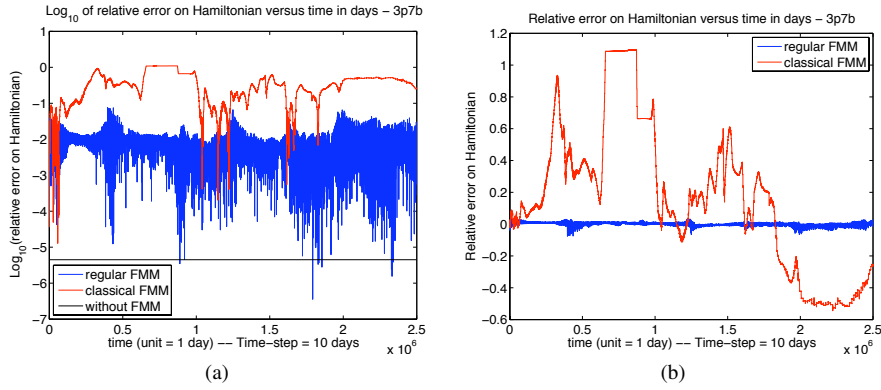
Fig. 7 Plot of the relative error on the Hamiltonian of the system, $L = 3$, $\mathcal{N}_L = 7$, $\mathcal{N}_o = 1$, $R_{reg} = 0.25$: (a) $\log_{10}$(relative error); (b) relative error.

of the trajectories and the displacement for one time-step. In this example, one can find that Uranus (resp. Jupiter) requires about 30,700 days (resp. 4,300) to describe one loop around the Sun. With a time-step equal to 10 days, the trajectory around the Sun will be described with about 3,070 locations (resp. 430) on a curve of length about 118 UA (resp. 32) which define a characteristic time-step length equal to 0.04 UA (resp. 0.07). Such reports give enough information to derive the choice of the FMM boxes.
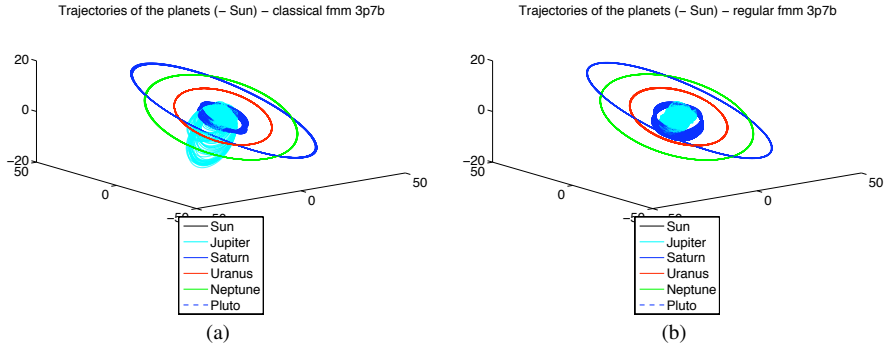
In the sequel, we present experiments for several values of the following parameters:

- ★ L: number of multipoles, truncature parameter in formulae (18) and (19). As is well known, the FMM expansion behaves like a geometric series with respect to L. A typical value for L is around 6, whereas a value around 15 or even 20 gives a very accurate approximation.
- ★ $\mathcal{N}_o$: order of neighborhood that defines the close and far interactions in the FMM oc-tree.
- ★ $\mathcal{N}_L$: number of levels of the oc-tree. We found in this experiment that a good tradeoff is $\mathcal{N}_L = 7$.
- ★ h_t: time-step size.
- ★ R_{reg}: ratio of the regularization zone on each side of a geometric group to the length of the geometric group. Example: For the 1D group $[0, 1]$, when $R_{reg} = 0.25$, the virtual corresponding group is $[-0.25, 1.25]$ and the regularization function operates on $[-0.25, 0.25]$ and $[-0.75, 1.25]$.
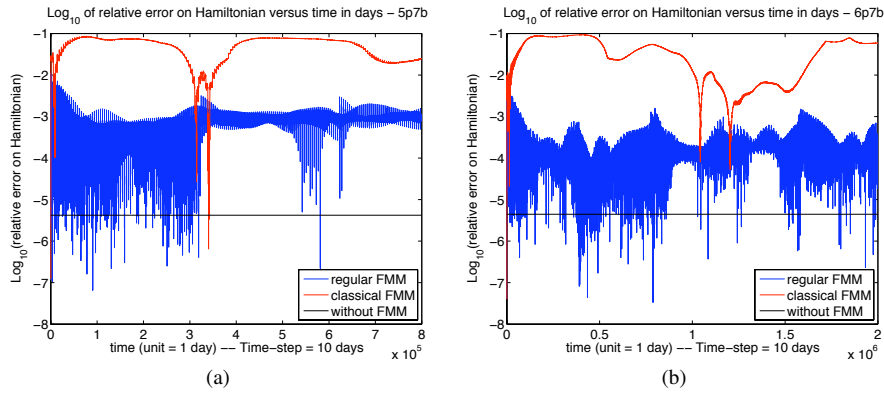
Figure 7-a shows the relative error (in log-scale) on the Hamiltonian versus time, for $L = 3$, $\mathcal{N}_L = 7$, $\mathcal{N}_o = 1$, $R_{reg} = 0.25$ and illustrates the impressive improvement brought by the regularization technique.

Figure 8 also shows a significant qualitative improvement, as far as trajectories are concerned. However, as is expected from a low-accuracy approximation of the potential, they are still quantitatively wrong.
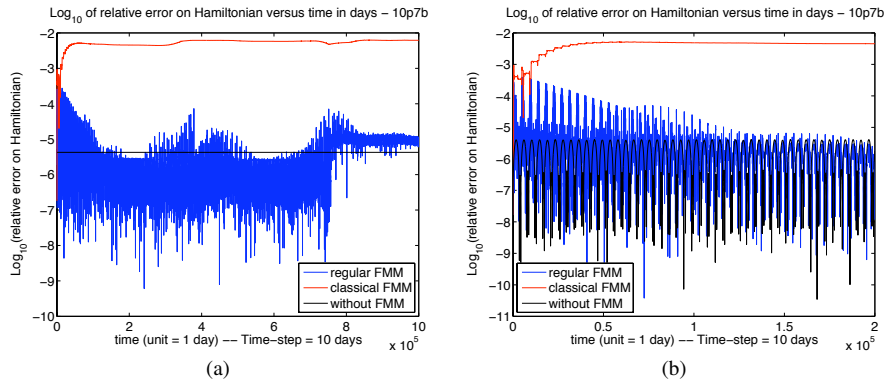
Figures 9 and 10 show that when L is increased, no gain in the qualitative behavior of the FMM is noticed. In contrast, the regularization technique enables energy preservation. For $L = 10$, the RFMM leads to accuracy comparable with what is obtained with the exact potential.

Trajectories of the planets (– Sun) – classical fmm 3p7b

Trajectories of the planets (– Sun) – regular fmm 3p7b

(a)

(b)

Fig. 8 Trajectories of the planets around the Sun, $L = 3$, using: (a) a classical FMM, (b) the regular FMM.

Log_{10} of relative error on Hamiltonian versus time in days – 5p7b

Log_{10} of relative error on Hamiltonian versus time in days – 6p7b

(a)

(b)

Fig. 9 Plot log_{10} of relative error on the Hamiltonian of the system, with: (a) $L = 5$, $\mathcal{N}_L = 7$, $\mathcal{N}_o = 1$, $R_{reg} = 0.25$; (b) $L = 6$, $\mathcal{N}_L = 7$, $\mathcal{N}_o = 1$, $R_{reg} = 0.25$.

Log_{10} of relative error on Hamiltonian versus time in days – 10p7b

Log_{10} of relative error on Hamiltonian versus time in days – 10p7b

(a)

(b)

Fig. 10 (a) Plot log_{10} of relative error on the Hamiltonian of the system, $L = 10$, $\mathcal{N}_L = 7$, $\mathcal{N}_o = 1$, $R_{reg} = 0.25$; (b) Zoom.

Trajectory of Jupiter – 6p7b   Trajectory of Saturn – 6p7b

(a)   (b)
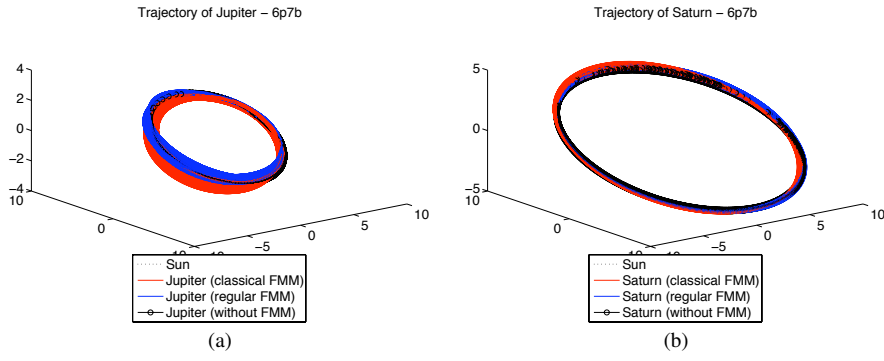
Fig. 11 Trajectories (with the different codes, $L = 6$) of the planets: (a) Jupiter; (b) Saturn.

(a)   (b)
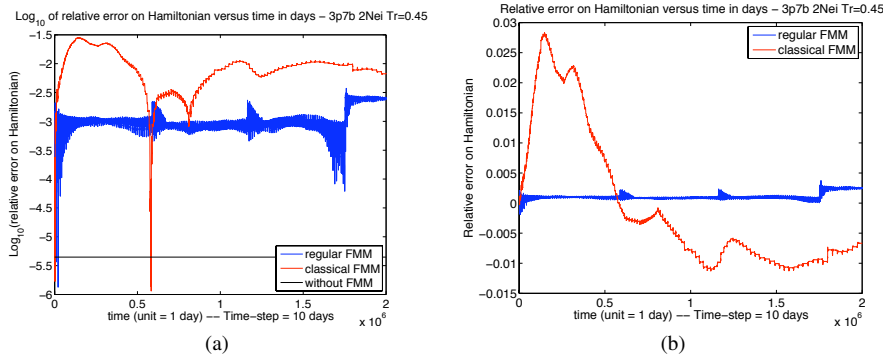
Fig. 12 Plot of the relative error on the Hamiltonian of the system, $L = 3$, $\mathcal{N}_L = 7$, $\mathcal{N}_o = 2$, $R_{reg} = 0.45$: (a) $\log_{10}$(relative error); (b) relative error.

In Figure 11, we show the trajectories of Jupiter and Saturn, using the three codes (with FMM, with RFMM or without FMM). We can see that the trajectories are more stable using the RFMM instead of the FMM.

In Figure 12, we consider an order of neighborhood $\mathcal{N}_o = 2$ and increase the ratio between the size of the regularization zone and the length of the geometric groups to $R_{reg} = 0.45$. Taking $L = 3$, we observe an error of order 10^{-3} on the Hamiltonian, comparable to the results obtained with $L = 5$ and $\mathcal{N}_o = 1$.

In Figure 13, we plot the trajectories of the planets both for the FMM and RFMM approximations, and observe a gain of stability in the regular case.

In terms of complexity of the algorithm, the observed overhead between FMM and RFMM is around 20% for $L = 3$, 10% for $L = 5$ and 6 and 1% for $L = 10$.
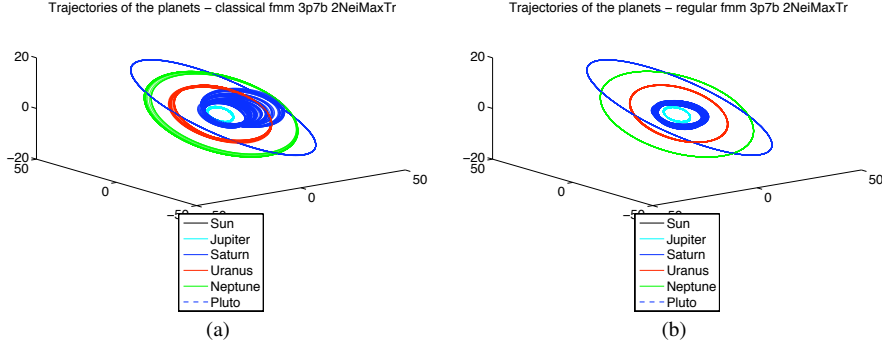
Fig. 13 Trajectories of the planets around the Sun, $L = 3$ and $\mathcal{N}_o = 2$, using: (a) a classical FMM, (b) the regular FMM.

4 Appendix

4.1 Details on the FMM approximation and FMM algorithm

The FMM approximation is based on the expansions given by the following results:

Result 1 (Multipole expansion): Assume that J source points $\{x_{j_1},...,x_{j_J}\}$ are contained in a group B_{src} of center C_{src} and of radius r. Let us denote by $(\rho_{j_p}, \theta_{j_p}, \phi_{j_p})$ the spherical coordinates of $(x_{j_p} - C_{src})$. Then for any x_i such that $\|x_i - C_{src}\|_2 > r$, denoting the spherical coordinates of $(x_i - C_{src})$ by $(\rho_{is}, \theta_{is}, \phi_{is})$, we have the expansion

$$\sum_{p=1}^{J} \frac{1}{\|x_i - x_{j_p}\|} q_{j_p} = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \frac{M_n^m}{\rho_{is}^{n+1}} Y_n^m(\theta_{is}, \phi_{is}) \tag{7}$$

with

$$M_n^m = \sum_{p=1}^{J} q_{j_p} \rho_{j_p}^n Y_n^{-m}(\theta_{j_p}, \phi_{j_p}).$$

and Y_n^m a spherical harmonic. The corresponding error estimate is

$$\left| \sum_{p=1}^{J} \frac{1}{\|x_i - x_{j_p}\|} q_{j_p} - \sum_{n=0}^{L} \sum_{m=-n}^{n} \frac{M_n^m}{\rho_{is}^{n+1}} Y_n^m(\theta_{is}, \phi_{is}) \right| \leq \frac{\sum_{p=1}^{J} |q_{j_p}|}{\rho_{is} - r} \left(\frac{r}{\rho_{is}} \right)^{L+1} \tag{8}$$

Result 2 (Conversion of a multipole expansion to a local expansion): Consider the J source points defined for Result 1. Let us consider that the target point x_i is contained in a group B_{trg} of center C_{trg} and radius r. Denoting by $(\rho_{st}, \theta_{st}, \phi_{st})$ the spherical coordinates of $(C_{src} - C_{trg})$ and by $(\rho_i, \theta_i, \phi_i)$ the spherical coordinates of $(x_i - C_{trg})$, under the condition that $\rho_{st} = \|C_{trg} - C_{src}\| > 2r$, the expansion given by (7) can be written

$$\sum_{p=1}^{J} \frac{1}{\|x_i - x_{j_p}\|} q_{j_p} = \sum_{l=0}^{\infty} \sum_{k=-l}^{l} L_l^k \rho_i^l Y_l^k(\theta_i, \phi_i) \tag{9}$$

with the translation operation

$$L_l^k = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \frac{M_n^m \iota^{|k-m|-|k|-|m|} A_n^m A_l^k Y_{l+n}^{m-k}(\theta_{st},\phi_{st})}{(-1)^n A_{l+n}^{m-k} \rho_{st}^{l+n+1}}. \tag{10}$$

and

$$A_n^m = \frac{(-1)^n}{\sqrt{(n-m)!(n+m)!}}.$$

We denote by $\mathscr{T}_{B_{trg}B_{src}}$ the translation operator which maps $(M_n^m)_{m,n}$ onto $(L_l^k)_{l,k}$ and write

$$(L_l^k)_{l,k} = \mathscr{T}_{B_{trg}B_{src}}(M_n^m)_{m,n} \tag{11}$$

The corresponding error estimate is

$$\left| \sum_{p=1}^{J} \frac{1}{\|x_i - x_{j_p}\|} q_{j_p} - \sum_{l=0}^{L} \sum_{k=-l}^{l} L_l^k \rho_i^l Y_l^k(\theta_i,\phi_i) \right| \le \frac{\sum_{p=1}^{J} |q_{j_p}|}{cr-r} \left(\frac{1}{c}\right)^{L+1} \tag{12}$$

with c satisfying $\rho_{st} > (c+1)r$.

The spherical harmonics are given from the associate Legendre functions P_n^m:

$$Y_n^m(\theta,\phi) = \sqrt{\frac{(n-|m|)!}{(n+|m|)!}} P_n^{|m|}(\cos\theta)e^{\iota m\phi}. \tag{13}$$

The associate Legendre functions can be calculated recursively thanks to the relations

$$\begin{cases} \left. \begin{array}{l} P_k^k(\cos\theta) = \frac{(2k)!}{2^k k!}(-\sin\theta)^k \\ P_{k+1}^k(\cos\theta) = (2k+1)\,\cos\theta\,P_k^k(\cos\theta) \end{array} \right\} \quad \forall k \ge 0, \\ \\ (l-k)P_l^k(\cos\theta) = (2l-1)\cos\theta P_{l-1}^k(\cos\theta) - (l+k-1)P_{l-2}^k(\cos\theta) \\ \qquad\qquad\qquad \forall l,k \,/\, 0 \le k \le l-2 \end{cases} \tag{14}$$

The derivation of the previous results is given in [12] and [13]. The definition of the special functions involved in those expansions and further details about their properties can be found in [16].

The first steps of the algorithm given in Subsection 2.1.1 can be written more precisely as follows:

- Step 0: Calculation of some q-independant quantities.
  - · Precalculation related to the translation operators $\mathscr{T}_{B_{trg}B_{src}}$.
  - · The far moments: $\forall B_{src} \in \mathscr{B}, \forall x_j \in B_{src}$, with $(x_j - C_{src}) \leftrightarrow (\rho_j,\theta_j,\phi_j)$

$$f_j^{l,k} = \rho_j^l Y_l^{-k}(\theta_j,\phi_j) \tag{15}$$

  - · The local moments: $\forall B_{trg} \in \mathscr{B}, \forall x_i \in B_{trg}$, with $(x_i - C_{trg}) \leftrightarrow (\rho_i,\theta_i,\phi_i)$

$$g_i^{l,k} = \rho_i^l Y_l^k(\theta_i,\phi_i) \tag{16}$$

- Step 1: Calculation of the far fields: $\forall B_{src} \in \mathscr{B}$,

$$F_{B_{src}}^{l,k} = \sum_{x_j \in B_{src}} q_j f_j^{l,k} \tag{17}$$

- Step 2: Calculation of the local fields – translations: $\forall B_{trg} \in \mathscr{B}$,

$$(G_{B_{trg}}^{l,k})_{l,k} = \sum_{B_{src} \text{ far from } B_{trg}} \mathscr{T}_{B_{trg} B_{src}}(F_{B_{src}}^{\lambda,\kappa})_{\lambda,\kappa} \tag{18}$$

- Step 3: Accumulation of the far interactions: $\forall B_{trg} \in \mathscr{B}$, $\forall x_i \in B_{trg}$,

$$(Aq)_i^{far} = \sum_{l=0}^{L} \sum_{k=-l}^{l} g_i^{l,k} G_{B_{trg}}^{l,k} \tag{19}$$

- Step 4: Calculation of the close interactions: $\forall B_{trg} \in \mathscr{B}$, $\forall x_i \in B_{trg}$,

$$(Aq)_i^{close} = \sum_{B_{src} \text{ close to } B_{trg}} \sum_{x_j \in B_{src}} \frac{1}{\|x_i - x_j\|} q_j \tag{20}$$

where L is the truncature parameter involved in the approximation of the multipole and local expansions (7) and (9).

4.2 Details on the FMM approximation of the gradient energy

The FMM approximation of the kernel $\nabla_x G(x,y)$ is derived from the FMM approximation of the kernel $G(x,y) = \frac{1}{\|x-y\|}$ by derivating the local moment $g_i^{l,k}$ w.r.t x_i, with the correspondance $(x_i - C_{trg}) \leftrightarrow (\rho_i, \theta_i, \phi_i)$ involved in (16). This derivation is performed below using the fact that the quantity $g_i^{l,k}$ is polynomial w.r.t $(x_i - C_{trg})$.

For $x = (x_1, x_2, x_3) \in \mathbb{R}^3$ of spherical coordinates (ρ, θ, ϕ), let us introduce the notation

$$\mathscr{E}_l^k(x_1, x_2, x_3) = \rho^l Y_l^k(\theta, \phi)$$

Thanks to the definition of the spherical harmonics (13), and the recursive relations on the associate Legendre functions (14), a simple exercise shows the following relations

$$
\begin{aligned}
\mathscr{E}_0^0(x_1, x_2, x_3) &= 1 \\
\mathscr{E}_l^l(x_1, x_2, x_3) &= (-x_1 - ix_2)^l \, \frac{\sqrt{(2l)!}}{2^l \, l!} \\
\mathscr{E}_{l+1}^l(x_1, x_2, x_3) &= x_3 \, \sqrt{2l+1} \, \mathscr{E}_l^l(x_1, x_2, x_3) \\
\mathscr{E}_{l+2}^k(x_1, x_2, x_3) &= C_l^k \, x_3 \, \mathscr{E}_{l+1}^k(x_1, x_2, x_3) + D_l^k \, \rho^2 \, \mathscr{E}_l^k(x_1, x_2, x_3) \\
\mathscr{E}_l^{-k}(x_1, x_2, x_3) &= \overline{\mathscr{E}_l^k(x_1, x_2, x_3)}
\end{aligned} \tag{21}
$$

with

$$C_l^k = \frac{2l+3}{\sqrt{(l+2+k)(l+2-k)}} \quad \text{and} \quad D_l^k = -\sqrt{\frac{(l+1+k)(l+1-k)}{(l+2+k)(l+2-k)}}$$

It comes directly the following relations for the gradient of $\mathscr{E}_l^k$

$$
\begin{aligned}
\nabla \mathscr{E}_0^0(x_1, x_2, x_3) &= 0 \\
\nabla \mathscr{E}_l^l(x_1, x_2, x_3) &= \tfrac{l}{(x_1+ix_2)} \, \mathscr{E}_l^l(x_1, x_2, x_3) \, V_1 \\
\nabla \mathscr{E}_{l+1}^l(x_1, x_2, x_3) &= \tfrac{\sqrt{2l+1}}{(x_1+ix_2)} \, \mathscr{E}_l^l(x_1, x_2, x_3) \, V_2
\end{aligned} \tag{22}
$$

with $V_1 = (1, i, 0)^T$ and $V_2 = (x_3 l, ix_3 l, x_1 + ix_2)^T$.

References

1. P. Chartier and E. Faou. Volume-energy preserving integrators for piecewise smooth approximations of Hamiltonian systems. M2AN, vol. 42 (2), pp. 223-241, 2008.
2. E. Hairer, C. Lubich and G. Wanner. Geometric Numerical Integration: structure-preserving algorithms for ordinary differential equations, Second edition, Springer, vol. 31, 2006.
3. B. Laird and B. Leimkuhler, A molecular dynamics algorithm for mixed hard-core/continuous potentials, Molecular Physics 98 (2000), 309-316.
4. Anne Kvaerno and Ben Leimkuhler, A time-reversible, regularized, switching integrator for the n-body problem, SIAM Journal on Scientific Computing 22 (2000), no. 3, 1016-1035.
5. K. N. Kudin and G. E. Scuseria. Range definitions for Gaussian-type charge distributions in fast multipole methods. J. Chem. Phys. 111 (6), pp. 2351-2356, aug. 1999.
6. J. C. Burant, M. C. Strain, G. E. Scuseria and M. J. Frisch. Analytic energy gradients for the Gaussian very fast multipole method (GvFMM). Chem. Phys. Lett., 248, pp. 43-49, jan. 1996.
7. M. C. Strain, G. E. Scuseria and M. J. Frisch. Achieving linear scaling for the electronic quantum Coulomb problem. Science, 271, pp. 51-53, jan. 1996.
8. C. A. White and M. Head-Gordon. Rotating around the quartic angular momentum barrier in fast multipole method calculation. J. Chem. Phys. 105 (12), pp. 5061-5067, sept. 1996.
9. Y. Shao, C. A. White and M. Head-Gordon. Efficient evaluation of the Coulomb force in density-functional theory calculations. J. Chem. Phys. 114 (15), pp. 6572-6577, april 2001.
10. C. A. White, B. G. Johnson, P. M.W. Gill and M. Head-Gordon. Linear scaling density functional calculations via the continuous fast multipole method. Chem. Phys. Lett., 253, pp. 268-278, may 1996.
11. C. A. White, B. G. Johnson, P. M.W. Gill and M. Head-Gordon. The continuous fast multipole method. Chem. Phys. Lett., 230, pp. 8-16, nov. 1994.
12. L. Greengard and V. Rokhlin. The Rapid Evaluation of Potential Fields in Three Dimensions. in Vortex Methods, Lecture Notes in Mathematics, 1360, Springer-Verlag, pp. 121-141, 1988.
13. L. Greengard and V. Rokhlin. A New Version of the Fast Multipole Method for the Laplace equation in the three dimensions. Acta Numerica, pp. 229-269, 1997.
14. J.A. Board and W.S. Elliot. Fast fourier transform accelerated fast multipole algorithm. Technical Report 94-001, Duke University Dept of Electrical Engineering, 1994.
15. H. G. Petersen, D. Soelvason, J. W. Perram and E. R. Smith. The very fast multipole method. J. Chem. Phys. 101 (10), pp. 8870-8876, nov. 1994.
16. M. Abramowitz and I. A. Stegun. Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, John Wiley, New York, 1972.